UNIVERSITY OF GLASGOW

COMPUTING DEPARTMENT

MULTUM OPERATING SYSTEM MEMO NO. 7. (16/7/73)

VIOS IMPLEMENTATION DESCRIPTION

1 Introduction.

The VIOS Implementation Description is intended to serve as reference documentation for the implementation and maintenance of the Virtual I/O System. Its scope is such as to include:

(a) global data structures;
(b) inter-program interfaces; and
(c) functional specifications of the component programs.

In other words, it describes VIOS as a system. Descriptions of individual components of VIOS are to be published separately.

An important objective of this document is to make information available as quickly as possible. Thus sections will be added as they become available, and new sections may revise topics treated in earlier sections. The result will be cumulative, rather than systematic. It is hoped that the benefits of early dissemination will outweigh the less convenient form in which the information is presented.

The most convenient binding for this document will probably be a loose-leaf folder.

2 Backing store data structures.

A catalog entry does not define the physical position of a file. Instead, a file is accessed indirectly, using its VTOC entry or descriptor. A catalog entry indicates which VTOC is to be examined and the number of the file's descriptor therein. This method has several advantages over direct addressing from the catalog entry:

(a) Because it contains no addressing data the catalog entry is very small, speeding up catalog searches. Access to the descriptor is fast, because direct.
(b) Files can be relocated or archived without changing catalogs (which may be recorded on off-line volumes).
(c) The implementation of links (synonyms) is simplified.
(d) The VTOC is indispensable to the implementation of effective procedures for recovery after catastrophic failure.

While a file and its parent catalog may be recorded on different volumes, a file must reside on the same volume as its VTOC entry. An exception to this rule occurs for magnetic tape files and for archival files which have been banished to some other volume.

When/

/When a volume is brought on-line access to its VTOC is gained via the Volume Identification Sector, recorded always on sector 0. The VIS is one component of the Volume Control File. On each volume the VCF supplies the primary administrative data.

One specially-designated volume, the root volume, must remain accessible at all times while the file system is active. This volume contains the SYSTEM catalog which is the root of the filestore tree and the parent of all user catalogs. Opening a user file may thus involve several steps:

(a) Searching SYSTEM for the parent catalog.
(b) Accessing the parent's VTOC entry.
(c) Searching the parent catalog.
(d) Accessing the file's VTOC entry.
(e) Accessing the file itself.

If the parent catalog is already open, steps (a) and (b) are omitted. If the file itself is already open, steps (c) and (d) are omitted. Under an Operating System it would be normal to leave a user's catalog open for the duration of his job.

2.1 The Volume Control File.

The first 48 sectors (2 tracks) of each volume are reserved by the file system to hold the Volume Control File (VCF). The VCF contains the Volume Identification Sector (VIS) and the Volume Allocation Map (VAM). Further components may be added in future extensions of the file system (e.g. a Volume Log). All of the remaining space (99.975% on a 9742) is available to the VTOC, catalogs and users' files.

The VCF is accessible as a file via the normal file system mechanisms.

2.1.1 The Volume Identification Sector.

The first sector of the VCF is always the VIS. This is used whenever a volume comes on-line to enable the VTOC (and thence user files) to be accessed. It keeps a record of the current status of the volume, and lists areas which have become unusable due to surface flaws. The individual fields of the VIS are now described.

The Volume Type indicates whether it is part of a filestore. If not, the rest of the VIS is undefined by the file system; if so, it also indicates whether the volume is public or private.

The (misnamed) Volume Serial Number is an alphanumeric identifier of 6 characters, beginning with a letter. The first 4 characters uniquely identify the filestore in which the volume belongs and are allocated centrally to prevent duplication. The last 2 characters uniquely identify the volume within its filestore, and always take the value "00" in the case of a root volume.

Some/

/Some defined period during a day is called a session (e.g. a work shift) and a volume which is normally on-line only during such a session is called sessional. Of course, a sessional volume may be brought on-line outside of its session if the need arises. When the session lasts all day the volume is said to be permanent (e.g. the root volume), while a volume whose session is of zero length is said to be a demand volume. The Time of Mounting field in the VIS defines the beginning of a volume's session and the Time of Demounting field defines the end. Both are given in minutes. As might be expected, they take the values 0 and 1440 (= 24 hr) for a permanent volume. For a demand volume, both fields are set to 0. The file system will allocate a file to a volume outwith the volume's session only if explicitly requested to do so.

Three fields of the VIS define the storage capacity of the volume. Track Size is given in sectors, Cylinder Size is given in tracks and Volume Size is given in cylinders.

A further 4 fields define how this storage is to be allocated. The Small Allocation Unit Size (SAU size) is given in sectors and must be an integral number of Basic Allocation Units. The Large Allocation Unit Size is given as an integral number of SAUs. So, too, are the Small Regime Size and the Large Regime Size. (The Basic Allocation Unit is a device-dependent quantum of storage: 6 sectors for the 9742 and 1 sector for the 9425/9427.)

The Position of SIL, Position of RPL and Position of VTOC fields are all given as SAU numbers within the volume (indexed from 0 at the beginning of the VCF). When the SIL or RPL is absent the corresponding field is set to -1 ([FFFF]).

The Latest File Reference # indicates the Reference # (unique integer) last assigned to a file on the volume at the time of the most recent volume checkpoint.

The Normal Shut-Down field is set to false when a volume is placed on-line. The file system sets it to true as the last act of the shut-down procedures executed when a volume is demounted. Thus if a volume comes on-line with this field false, it is clear that it has not been demounted normally (perhaps as the result of a system failure).

The # of Flawed SAUs field counts the SAUs on the volume where unrecoverable parity failures (assumed due to surface flaws) have been detected. The Flaw List gives the positions of these SAUs, along with the positions of their replacements.

The Filler is a field reserved for extensions to the file system.

See Figure 2.1.

Figure 2.1(a) : VIS Type Definition.

```
type VIS  =  record
               Volume Type  :  (nonfs, private, public);
               Volume Serial Number : array [1..6] of char;
               Time of Mounting, Time of Demounting
                                    : integer;
               Track Size, Cylinder Size, Volume Size : integer;
               Small Allocation Unit Size, Large Allocation
                 Unit Size        : integer;
               Small Regime Size, Large Regime Size : integer;
               Position of SIL, Position of RPL,
               Position of VTOC : integer;
               Latest File Reference #: longinteger;
               Normal Shut-Down          : Boolean;
               # of Flawed SAUs          : integer;
               Flaw List
                         : array [1..50] of
                                record
                                Flaw, Replacement : integer
                                end;
               Filler  : array [1..8] of integer
               end
```

Figure 2.1(b) : VIS Layout.

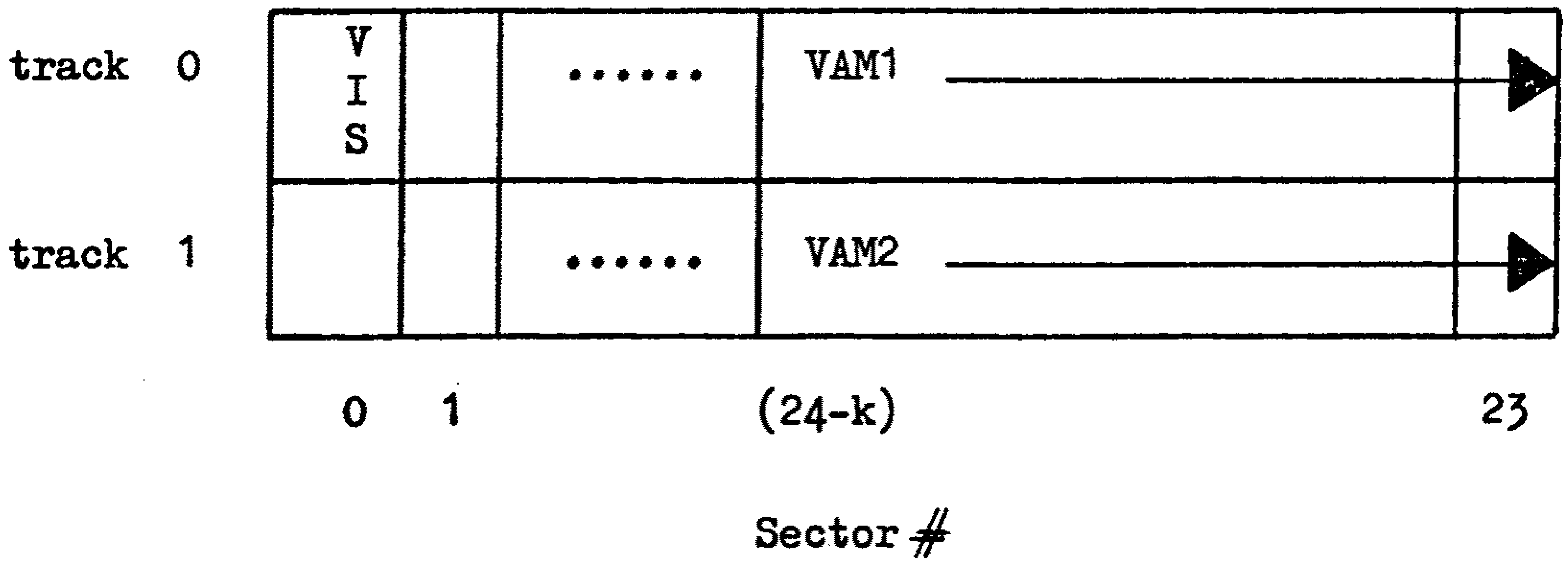
Position		Length
word 0	Volume Type	1 word
1	Volume Serial Number	3
4	Time of Mounting	1
5	Time of Demounting	1
6	Track Size	1
7	Cylinder Size	1
8	Volume Size	1
9	Small Allocation Unit Size	1
10	Large " " "	1
11	Small Regime Size	1
12	Large Regime Size	1
13	Position of SIL	1
14	" " RPL	1
15	" " VTOC	1
16	Latest File Reference #	2
18	Normal Shut-Down	1
19	# of Flawed SAUs	1
20	Flaw List	100
120	Filler	8

2.1.2 The Volume Allocation Map.

The allocation status of each SAU on a volume is indicated in its <u>Volume Allocation Map</u> (VAM). The VAM has two components, VAM1 and VAM2, each a bit map of the entire storage space in which one SAU is represented by one bit. Thus their lengths are given (in sectors) by:

$$k = \left\lceil \frac{\text{Track Size} \times \text{Cylinder Size} \times \text{Volume Size}}{2048 \times \text{SAU Size}} \right\rceil$$

VAM1 is stored in the final sectors of track 0 and VAM2 in the corresponding position of track 1, thus:



The first two words of both VAM1 and VAM2 are used to hold a checksum of that part. Word 2 of VAM1 points to the word at which the next search for free space should begin, while word 3 gives the number of free SAU. Words 2 and 3 of VAM2 are not used.

VAM1 is used to determine which SAUs are part of some file and hence not eligible for re-allocation. A 0-bit denotes an unused SAU, a 1-bit denotes a SAU which is in use. If it appears from VAM1 that a SAU is available, VAM2 is consulted to confirm this, or to determine that the SAU belonged to a file which has been deleted or contracted since the last volume checkpoint. In the latter case the SAU cannot yet be re-allocated. For VAM2 a 0-bit denotes "available".

When a volume checkpoint takes place, the administrative data is brought up-to-date <u>on the volume</u>, VAM1 is checksummed, and VAM2 is cleared.

2.2 The Volume Table of Contents.

The <u>Volume Table of Contents</u> (VTOC) is an array of <u>descriptors</u>. Each descriptor occupies one sector and is either empty or contains some part of the address-mapping and other administrative data maintained for each file allocated on the volume. In every VTOC the first 5 descriptors are assigned as follows:

0 : the VCF,
1 : the VTOC itself,
2 : the SIL (where present),
3 : the RPL (" "),
4 : the head of a doubly-linked list of empty descriptors.

The empty-descriptor list is chained through the "father" and "son" pointers (see below). Descriptors 2 and 3 are never placed on the empty-descriptor list, even if no SIL or RPL is currently allocated on the volume. Whether a descriptor is empty or not is indicated by the Descriptor Class field.

The three following fields give the Local Name, Catalog Name and Account name (Accnt) pertaining to the file.

Status Now indicates all the modes in which the file is currently open.

The Creation Date and Expiry Date fields have the obvious significance and are expressed in days relative to January 1, 1973 (which is day 0).

Current Size and Maximum Size, both expressed as highest SAU numbers, define the amounts of storage which have been, and can be, allotted to the file. The Current Size and Highest Sector fields together indicate the (virtual) file address of the highest numbered sector to which data has been written - in other words, the end-of-file position. Maximum Size is negated for files belonging to the Small Regime.

The Father and Son fields point to the descriptors for the next-older and next-younger generations of the file. (If the descriptor is empty they point to the previous and next descriptors respectively, on the free descriptor chain.) A null pointer is indicated by a value of -1 ([FFFF]).

The File Reference # is a 32-bit integer assigned to the file when it was created and is unique on that volume. The Creation Time is the time of day at which the file was created, expressed in seconds as a 32-bit integer.

The Access Method Data is reserved for the use of access methods.

Filler 1 is reserved for extensions to the file system.

The Sclass field indicates the storage class of the file (temporary, archival, permanent, immobile, magtape and temporary magtape). The Reels case specifies the tape reels occupied by a magnetic tape file. Otherwise, Extents defines a file's direct-access storage and Backup defines its dump/archival copies. In an Extents record, VLSAU denotes the virtual (file) address of the last SAU included in the extent, while PFSAU denotes the physical (device) address of the first SAU in the extent. Both quantities are expressed as SAU numbers indexed from 0.

Filler 2 and Filler 3 are reserved for extensions to the file system.

See Figure 2.2.

Figure 2.2(a) : Descriptor Type Definition.

```
type Descriptor  =  record
                      Descriptor Class     : (empty, primary, ......);
                      Local Name           : array [1..16] of char;
                      Catalog, Accnt       : array [1..12] of char;
                      Status Now           : Access;
                      Creation Date, Expiry Date, Current Size,
                      Maximum Size, Highest Sector
                                           : integer;
                      Father, Son          : integer;
                      File Reference #,    Creation Time
                                           : longinteger;
                      Access Method Data   : array [1..16] of char;
                      Filler 1             : array [1..12] of char;
                      case sclass : fclass of
                      ...... disc :
                          (Extents : array [1..16] of
                                     record
                                       PFSAU, VLSAU : integer
                                     end;
                          Backup   :   array [1..64] of char;
                          Filler 2 :   array [1..32] of char);
                      .... magtape :   (Reels : array [1..128] of char;
                                       Filler 3: array [1..32]  of char)
                      end
```

Figure 2.2(b) : Descriptor Layout.

Position	Field	Field	Length	Length
word 0	Descriptor Class		1 word	
1	Local Name		8	
9	Catalog		6	
15	Accnt		6	
21	Status Now		1	
22	Creation Date		1	
23	Expiry Date		1	
24	Current Size		1	
25	Maximum Size		1	
26	Highest Sector		1	
27	Father		1	
28	Son		1	
29	File Reference #		2	
31	Creation Time		2	
33	Access Method Data		8	
41	Filler 1		5	
47	Sclass		1	
48	Extents	Reels	32	64
80	Backup		32	
112	Filler 2	Filler 3	16	

2.3 Catalogs.

The first sector of a catalog is reserved for administrative purposes, and may eventually contain an index to assist catalog searches as well as file system resource allocation data. Initially, the catalogs will be searched exhaustively and there will not be any resource controls. All of the remaining sectors can be used to hold catalog entries, there being one entry per file (assuming the absence of links to files in the same catalog). A catalog entry occupies just 64 bytes, so that 4 entries can be stored in a sector.

2.3.1 Catalog entries.

An entry may be vacant, or contain data pertaining to some file. These cases are distinguished by the <u>Entry Status</u> field. The <u>Local Name</u> of the entry comprises a 12-character <u>Short Name</u> and a 4-character <u>File Type</u>. Space occupied by a file is chargeable to the account given in the <u>Account Name</u> field. The <u>FRNumber</u> is the 32-bit identification number of the file, as recorded in its VTOC entry. <u>MaxGens</u> and <u>CurrGens</u> give respectively the maximum number and the current number of <u>previous</u> generations of the file, such that:

$$0 \leqslant \text{MaxGens} \leqslant 9 \quad \text{and}$$

$$0 \leqslant \text{CurrGens} \leqslant \text{MaxGens}.$$

The <u>Entry Type</u> distinugishes original catalog entries from those which merely link to the descriptors of files whose original catalog entry is elsewhere. An array of four 'Access' codes, <u>Allowed</u>, defines the access permitted to the <u>entry</u>'s owner, his partners, the public, and the most privileged of the users listed in the file's descriptor(s).

The position of an entry's descriptor(s) is given by <u>Descr1</u>, <u>Descr2</u> and <u>Volume Serial</u>. The first two are the indices of the descriptors of the newest and oldest generations respectively, while the third is the alphanumeric identification of their volume.

<u>Filler</u> occupies 5 words and is reserved for extensions to the file system.

See Figure 2.3.

Figure 2.3(a) : Catalog Entry Type Definition.

```
type Catalog Entry  =
        record
              Entry Status      :  (vacant, filled);
              Local Name        :
                record
                      Short Name  :  array [1..12] of char;
                      File Type   :  array [1..4]  of char
                end;
              Account Name      :  array [1..12] of char;
              FRNumber          :  longinteger;
              MaxGens, CurrGens
                                :  integer;
              Entry Type        :  (File Entry, Link Entry);
              Allowed           :  array [1..4] of Access;
              Descr1, Descr2    :  integer;
              Volume Serial     :  array [1..6] of char;
              Filler            :  array [1..10] of char
        end
```

Figure 2.3(b) : Catalog Entry Layout.

Position		Length
word 0	Entry Status	1 word
1	Local Name	8
9	Account Name	6
15	FR Number	2
17	Max Gens	1
18	Curr Gens	1
19	Entry Type	1
20	Allowed	2
22	Descr 1	1
23	Descr 2	1
24	Volume Serial	3
27	Filler	5